IEEE Standards

IEEE Std 1516.3™-2003

1516.3™

IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP)

IEEE Computer Society

Sponsored by the
Simulation Interoperability Standards Committee

IEEE

Published by
The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

23 April 2003

Print: SH95088
PDF: SS95088

IEEE Std 1516.3™-2003

IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP)

Sponsor

Simulation Interoperability Standards Committee
of the
IEEE Computer Society

Approved 20 March 2003

IEEE-SA Standards Board

Abstract: The processes and procedures that should be followed by users of the High Level Architecture (HLA) to develop and execute federations are defined in this recommended practice. This recommended practice is not intended to replace low-level management and systems engineering practices native to HLA user organizations, but is rather intended as a higher-level framework into which such practices can be integrated and tailored for specific uses.
Keywords: federate, federation, Federation Development and Execution Process (FEDEP), High Level Architecture (HLA), Object Model (OM), runtime infrastructure (RTI)

The Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York, NY 10016-5997, USA

Print: ISBN 0-7381-3584-4 SH95088
PDF: ISBN 0-7381-3585-2 SS95088

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

Use of an IEEE Standard is wholly voluntary. The IEEE disclaims liability for any personal injury, property or other damage, of any nature whatsoever, whether special, indirect, consequential, or compensatory, directly or indirectly resulting from the publication, use of, or reliance upon this, or any other IEEE Standard document.

The IEEE does not warrant or represent the accuracy or content of the material contained herein, and expressly disclaims any express or implied warranty, including any implied warranty of merchantability or fitness for a specific purpose, or that the use of the material contained herein is free from patent infringement. IEEE Standards documents are supplied "**AS IS**."

The existence of an IEEE Standard does not imply that there are no other ways to produce, test, measure, purchase, market, or provide other goods and services related to the scope of the IEEE Standard. Furthermore, the viewpoint expressed at the time a standard is approved and issued is subject to change brought about through developments in the state of the art and comments received from users of the standard. Every IEEE Standard is subjected to review at least every five years for revision or reaffirmation. When a document is more than five years old and has not been reaffirmed, it is reasonable to conclude that its contents, although still of some value, do not wholly reflect the present state of the art. Users are cautioned to check to determine that they have the latest edition of any IEEE Standard.

In publishing and making this document available, the IEEE is not suggesting or rendering professional or other services for, or on behalf of, any person or entity. Nor is the IEEE undertaking to perform any duty owed by any other person or entity to another. Any person utilizing this, and any other IEEE Standards document, should rely upon the advice of a competent professional in determining the exercise of reasonable care in any given circumstances.

Interpretations: Occasionally questions may arise regarding the meaning of portions of standards as they relate to specific applications. When the need for interpretations is brought to the attention of IEEE, the Institute will initiate action to prepare appropriate responses. Since IEEE Standards represent a consensus of concerned interests, it is important to ensure that any interpretation has also received the concurrence of a balance of interests. For this reason, IEEE and the members of its societies and Standards Coordinating Committees are not able to provide an instant response to interpretation requests except in those cases where the matter has previously received formal consideration.

Comments for revision of IEEE Standards are welcome from any interested party, regardless of membership affiliation with IEEE. Suggestions for changes in documents should be in the form of a proposed change of text, together with appropriate supporting comments. Comments on standards and requests for interpretations should be addressed to:

Secretary, IEEE-SA Standards Board
445 Hoes Lane
P.O. Box 1331
Piscataway, NJ 08855-1331
USA

Note: Attention is called to the possibility that implementation of this standard may require use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. The IEEE shall not be responsible for identifying patents for which a license may be required by an IEEE standard or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Authorization to photocopy portions of any individual standard for internal or personal use is granted by the Institute of Electrical and Electronics Engineers, Inc., provided that the appropriate fee is paid to Copyright Clearance Center. To arrange for payment of licensing fee, please contact Copyright Clearance Center, Customer Service, 222 Rosewood Drive, Danvers, MA 01923 USA; +1 978 750 8400. Permission to photocopy portions of any individual standard for educational classroom use can also be obtained through the Copyright Clearance Center.

Introduction

[This introduction is not part of IEEE Std 1516.3-2003, IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP).]

The High Level Architecture (HLA) has been designed to facilitate interoperability among simulations and to promote reuse of simulations and their components. The HLA is composed of three major components:

— *HLA rules:* A set of ten basic rules that together describe the general principles defining the HLA.
— *HLA interface specification:* A description of the functional interface between simulations (federates) and the HLA runtime infrastructure (RTI).
— *HLA Object Model Template (OMT):* A specification of the common format and structure for documenting HLA object models.

In an HLA application, any number of physically distributed simulation systems can be brought together into a unified simulation environment to address the needs of new applications. These types of environments are known as HLA federations. The HLA specifications together define an overarching framework for the construction and execution of federations.

Within the various government and commercial organizations that comprise the HLA community, many different approaches to project management and systems engineering are being used. Such practices, procedures, and methodologies have evolved over time based on how well they serve the different functional areas and user communities for which they are intended. Many of these approaches currently use modeling and simulation (M&S) as a key enabler of certain functions, such as concept evaluation, testing, and training. However, a significant number of organizations adopting the HLA have not yet determined how to tailor their native management and engineering processes to take advantage of the HLA. For instance, while many in the analysis community have established procedures for non-runtime exchange of data from one simulation to another, the opportunities provided by the HLA for more dynamic exchange of data at runtime require that existing engineering processes be modified or augmented in order to take advantage of such opportunities. Even in communities in which distributed simulation is more commonplace (e.g., training), migration to the HLA generally requires some modification to existing management and engineering processes to capture the benefits offered by the HLA. As simulation users perform this migration, it is critical that guidance be available to orient new users to the specific set of tasks and activities necessary to develop and execute HLA federations.

This recommended practice describes the HLA FEDEP. The purpose of this recommended practice is to describe a generalized process for building and executing HLA federations. It is not intended to replace the existing management and systems design/development methodologies of HLA user organizations, but rather to provide a high-level framework for HLA federation construction and execution into which other systems engineering practices native to each individual application area can be easily integrated. In addition, the HLA FEDEP is not intended to be prescriptive, in that it does not specify a "one size fits all" process for all HLA users. Rather, the FEDEP defines a generic, common sense systems engineering methodology for HLA federations that can and should be tailored to meet the needs of user applications.

Although every HLA application requires a basic agreement among all federates as to the systems engineering approach that will be used to develop and execute the federation, there can be significant variability in the degree of formality defined in the chosen process. The primary driver for how much formality is required is typically the size and complexity of the application. For example, in large complex federations, requirements and associated schedules for delivery of federation products are generally very explicit, as is the content and format for documentation of these products. In smaller or less complex applications, a less structured process with fewer constraints on the types, formats, and content of federation products may be perfectly reasonable and may have certain efficiency advantages as compared to a more formalized process.

Other secondary factors may also influence how the FEDEP is tailored (or adapted) for a specific application. For instance, some communities may have documentation requirements that are unique to their application area. In this case, the activities required to produce these products must be accounted for in the overall process. The reuse potential of these and other required federation products may also influence the nature and formality of the activities that produce them. Another factor is the availability of reusable federation products and persistent federation development teams as opportunities for shortcuts, whereby it may be possible to identify and take advantage of a more streamlined, efficient development process. Finally, practical resource constraints (i.e., cost, schedule) may dictate how certain activities are performed and how the associated federation products are produced and documented.

In summary, it is recognized that the needs and requirements of the distributed simulation community are quite diverse. The HLA provides a generalized architecture for simulation interoperability; however, strict adherence to the HLA specifications is not, by itself, sufficient to ensure a fully consistent, interoperable distributed simulation environment. For instance, issues such as the need for consistent environmental databases and for consistent behavior representations of objects modeled by more than one federate are critical to achieving interoperability; however, these types of issues cannot be fully addressed solely through adherence to the HLA specifications. Although some technical or managerial issues may be unique to a given application, many other issues associated with building and executing a fully interoperable HLA federation are more general in nature. The HLA FEDEP is offered to the HLA community as a starting framework for identifying and addressing these more general issues, as discussed within the context of a full end-to-end process model for the development and execution of distributed simulation environments (federations) that fully conform with the HLA specifications. This framework can and should be tailored as appropriate to address the unique issues, requirements, and practical constraints of each individual application. It is expected that this framework will provide a viable foundation for all HLA applications and will assist the users in defining the specific tasks and activities necessary to support their particular needs.

Participants

At the time this recommended practice was completed, the working group had the following membership:

Robert Lutz, *Chair*
Reed Little, *Vice Chair*
Katherine L. Morse, *Secretary*

Kevin Alix
Tobias Almén
Matt Aylward
Jane Bachman
Mike Bachmann
Cynthia Ballard
Geoff Barbier
William Beavin
Brian Beebe
Jeff Bergenthal
Richard Bernstein
Ronald Bertin
Donna Blake
Bob Bolling
Jake Borah
Steve Boswell
Christina Bouwens
Karsten Brathen
Tom Brown
Carl Byers
Chris Catotti
Bob Chapman
Anders Christensen
Doug Clark
Will Clayton
Mark Crooks
Rick Davenport
Lloyd Dean
Arif Dhanidina
Vic DiRienzo
Uwe Dobrindt
Richard Dobson
Ray Drake
Tedd Dugone
Gary England
Jeffrey Farber
John Fay
Francis Feldpausch
Jason Ferreira
Frank Fiedrich
Keith Ford
Masa Furuichi
Peter Gabrielsson
Steve Galloway
Allan R. Gettman
Dave Gill
Daniel Girardot
Jean Graffagnini
Allison Griffin
Paul Gustavson
Hans-Peter Gustavsson
Rüdiger Gutte
Furman Haddix
Susan Harkrider
David Hayward
Jeremy Hendrix
Jim Hollenbach
Gilbert Hu
Wim Huiskamp
Jean-Louis Igarza
Magnus Johnsson
Michel Keuning
Erik de Koster
Joe Lacetera
John Langworthy
Kaj Lehtovaara
Arjan Lemmers
Mike Lightner
Staffan Löf
Björn Löfstrand
Franklin Lu
Aykut Manioglu
Ted Marz
Anders Mattson
Michael McAndrew
Mark McCall
Hans-Peter Menzler
Jack Michalek
Michael Myjak
Eckehard Neugebauer
Anna Niklasson
Tommy Nordqvist
Eileen O'Donnell
Peter Ohlander
Gunnar Öhlund
Jennifer Park
John Pearson
Colin Petford
Mikel Petty
Pascal Peyronnet
Kay Pixius
Vic Polkowski
Jonathan Prescott
Balajee Ramakrishnananda
Richard Reading
Godfrey Rhodes
John Roberts
Maria Rossi
Chris Rougêt
Randy Saunders
Judy Schandua
Thomas Schmatz
Aaron Schwartzbard
Roy Scrudder
Rick Severinghaus
Mike Shafto
Steven Sheasby
Carl Simchick
Joseph Steel
Allen Stotz
Marcy Stutzman
Robert Swift
Simon Taylor
Georgios Theodoropoulos
W. H. Thomas
Chris Turrell
Hartmut Ufer
Branislav Vakos
Chin Keng Wai
Bill Waite
Jim Watkins
Mike Whitehorn
John Woodyard
Philomena Zimmerman

The following members of the balloting committee voted on this recommended practice. Balloters may have voted for approval, disapproval, or abstention.

Brian Beebe
Emmet Beeker
Jeff Bergenthal
Mitchell Bonnett
Jake Borah
Christina Bouwens
Marco de Vos
Dr. Guru Dutt Dhingra
Uwe Dobrindt
Ray Drake
Dr. Sourav Dutta
Brian Goldiez
Ronald Hofer
James Ivers
Pi-Cheng Law
Kaj Lehtovaara
Reed Little
Robert Lutz
Faramarz Maghsoodlou
Ted Marz
Mark McAuliffe
Katherine L. Morse
Diane Moss
Tom Mullins
Chris Rougêt
Randy Saunders
Terry Schmidt
Roy Scrudder
Graham Shanks
Steven Sheasby
Donald Shillcutt
Susan Solick
William Tucker
Guy Zimmer
Philomena Zimmerman
William Zimmerman

When the IEEE-SA Standards Board approved this recommended practice on 20 March 2003, it had the following membership:

Don Wright, *Chair*
Howard M. Frazier, *Vice Chair*
Judith Gorman, *Secretary*

H. Stephen Berger
Joe Bruder
Bob Davis
Richard DeBlasio
Julian Forster*
Toshio Fukuda
Arnold M. Greenspan
Raymond Hapeman
Donald M. Heirman
Laura Hitchcock
Richard H. Hulett
Anant Jain
Lowell G. Johnson
Joseph L. Koepfinger*
Tom McGean
Steve Mills
Daleep C. Mohla
William J. Moylan
Paul Nikolich
Gary Robinson
Malcolm V. Thaden
Geoffrey O. Thompson
Doug Topping
Howard L. Wolfman

*Member Emeritus

Also included are the following nonvoting IEEE-SA Standards Board liaisons:

Alan Cookson, *NIST Representative*
Satish K. Aggarwal, *NRC Representative*

Andrew Ickowicz
IEEE Standards Project Editor

Contents

IEEE Recommended Practice for High Level Architecture (HLA) Federation Development and Execution Process (FEDEP)

1. Overview

1.1 Scope

This recommended practice defines the processes and procedures that should be followed by users of the High Level Architecture (HLA) to develop and execute federations. It is not intended to replace low-level management and systems engineering practices native to HLA user organizations, but is rather intended as a higher-level framework into which such practices can be integrated and tailored for specific uses.

1.2 Purpose

The HLA has been designed to be applicable across a wide range of functional applications. The purpose of this architecture is to facilitate interoperability among simulations and promote reuse of simulations and their components.

A named set of applications (e.g., simulations, loggers, viewers) interacting via the services of the HLA runtime infrastructure (RTI) in accordance with a common rule set and a common HLA Object Model (OM) is known as an HLA *federation*. The purpose of this recommended practice is to describe a high-level process by which HLA federations can be developed and executed to meet the needs of a federation user or sponsor. It is expected that the guidelines provided in this recommended practice are generally relevant to and can facilitate the development of most HLA federations.

2. References

The three specifications that together compose the HLA provide the technical foundation for designing and developing all HLA federations. These specifications are described in the following documents:

HLA IEEE 1516 version

IEEE Std 1516™-2000, IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)—Framework and Rules.[1, 2]

IEEE Std 1516.1™-2000, IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)—Federate Interface Specification.

[1]The IEEE products referred to in this clause are trademarks belonging to the Institute of Electrical and Electronics Engineers, Inc.

[2]IEEE publications are available from the Institute of Electrical and Electronics Engineers, 445 Hoes Lane, P.O. Box 1331, Piscataway, NJ 08855-1331, USA (http://standards.ieee.org/).

IEEE Std 1516.2™-2000, IEEE Standard for Modeling and Simulation (M&S) High Level Architecture (HLA)—Object Model Template (OMT) Specification.

HLA Version 1.3[3]

High-Level Architecture Rules, Version 1.3, U. S. Department of Defense, Apr. 1998.
A set of ten basic rules that together describe the general principles defining the HLA.

High Level Architecture Interface Specification, Version 1.3, U. S. Department of Defense, Apr. 1998.
A description of the functional interface between simulations (federates) and the HLA RTI.

High-Level Architecture Object Model Template (OMT) Specification, Version 1.3, U. S. Department of Defense, Apr. 1998.
A specification of the common format and structure for documenting HLA OMs.

3. Definitions, abbreviations, and acronyms

3.1 Definitions

The following terms and definitions, as drawn from the IEEE 1516 series of HLA specifications and other relevant sources (see Annex A—Bibliography), shall apply throughout this recommended practice.

3.1.1 conceptual model: An abstraction of the real world that serves as a frame of reference for federation development by documenting simulation-neutral views of important entities and their key actions and interactions. The federation conceptual model describes what the federation will represent, the assumptions limiting those representations, and other capabilities needed to satisfy the user's requirements. Federation conceptual models are bridges between the real world, requirements, and design.

3.1.2 federate: An application that may be or is currently coupled with other software applications under a Federation Object Model Document Data/Federation Execution Data (FDD/FED) and a runtime infrastructure (RTI). This may include federation managers, data collectors, real world ("live") systems (e.g., C4I systems, instrumented ranges, sensors), simulations, passive viewers, and other utilities.

3.1.3 federation: A named set of federate applications and a common Federation Object Model (FOM) that are used as a whole to achieve some specific objective.

3.1.4 Federation Object Model (FOM): A specification defining the information exchanged at runtime to achieve a given set of federation objectives. This includes object classes, object class attributes, interaction classes, interaction parameters, and other relevant information.

3.1.5 Federation Object Model (FOM) Document Data (FDD): The data and information in a FOM document that is used by the Create Federation Execution service to initialize a newly created federation execution. Also called "Federation Execution Data," or "FED," in HLA Version 1.3 specifications.

3.1.6 runtime infrastructure (RTI): The software that provides common interface services during a High Level Architecture (HLA) federation execution for synchronization and data exchange.

[3]HLA Version 1.3 specifications are available at http://www.dmso.mil.

3.1.7 scenario: (A) Description of an exercise. It is part of the session database that configures the units and platforms and places them in specific locations with specific missions (see IST-SP-96-01 [B3][4]). **(B)** An initial set of conditions and time line of significant events imposed on trainees or systems to achieve exercise objectives (see *The Authoritative Dictionary of IEEE Standards Terms*, Seventh Edition [B2]).

3.1.8 Simulation Object Model (SOM): A specification of the types of information that an individual federate could provide to High Level Architecture (HLA) federations as well as the information that an individual federate can receive from other federates in HLA federations. The standard format in which SOMs are expressed facilitates determination of the suitability of federates for participation in a federation.

3.2 Abbreviations and acronyms

The following abbreviations and acronyms pertain to this recommended practice.

BOM	Base Object Model
CASE	Computer-Aided Software Engineering
COTS	commercial off-the-shelf
CPU	Central Processing Unit
FDD	FOM Document Data
FED	Federation Execution Data
FEDEP	Federation Development and Execution Process
FOM	Federation Object Model
GOTS	government off-the-shelf
HLA	High Level Architecture
IEEE	Institute of Electrical and Electronics Engineers, Inc.
LAN	Local Area Network
M&S	modeling and simulation
MOM	Management Object Model
OM	Object Model
OMT	Object Model Template
RTI	runtime infrastructure
SOM	Simulation Object Model
VV&A	verification, validation, and accreditation
WAN	Wide Area Network

4. FEDEP model: top-level view

One of the design goals identified early in the development of the HLA was the need for a high degree of flexibility in the process by which HLA applications could be composed and executed to achieve the objectives of particular applications. Because of this basic desire to avoid mandating unnecessary constraints on how HLA applications are constructed and executed, it was recognized that the actual process used to develop and execute HLA federations could vary significantly within or across different user applications. For instance, the types and sequence of low-level activities required to develop and execute analysis-oriented federations is likely to be quite different from those required to develop and execute distributed training exercises. However, at a more abstract level, it is possible to identify a sequence of seven very basic steps that all HLA federations should follow to develop and execute their federations. Figure 1 illustrates each of these steps and is summarized as follows:

[4]The numbers in brackets correspond to those of the bibliography in Annex A.

— *Step 1: Define federation objectives.* The federation user, the sponsor, and the federation development team define and agree on a set of objectives and document what must be accomplished to achieve those objectives.
— *Step 2: Perform conceptual analysis.* Based on the characteristics of the problem space, an appropriate representation of the real world domain is developed.
— *Step 3: Design federation.* Existing federates that are suitable for reuse are identified, design activities for federate modifications and/or new federates are performed, required functionalities are allocated to the federates, and a plan is developed for federation development and implementation.
— *Step 4: Develop federation.* The Federation Object Model (FOM) is developed, federate agreements are established, and new federates and/or modifications to existing federates are implemented.
— *Step 5: Plan, integrate, and test federation.* All necessary federation integration activities are performed, and testing is conducted to ensure that interoperability requirements are being met.
— *Step 6: Execute federation and prepare outputs.* The federation is executed and the output data from the federation execution is pre-processed.
— *Step 7: Analyze data and evaluate results.* The output data from the federation execution is analyzed and evaluated, and results are reported back to the user/sponsor.

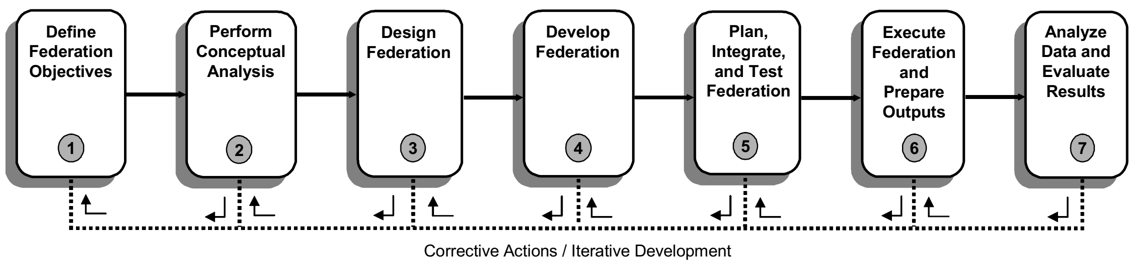

Figure 1—Federation development and execution process (FEDEP), top-level view

Since this seven-step process can be implemented in many different ways depending on the nature of the application, it follows that the time and effort required to build and execute an HLA federation can also vary significantly. For instance, it may take a federation development team several weeks to fully define the real world domain of interest for very large, complex applications. In smaller, relatively simple applications, the same activity could potentially be conducted in a day or less. Differences in the degree of formality desired in the process can also lead to varying requirements for federation resources.

Personnel requirements can also vary greatly depending on the scope of the federation application. In some situations, highly integrated teams composed of several individuals may be needed to perform a single role in a large, complex federation, while a single individual may perform multiple roles in smaller applications. Examples of the types of roles individuals can assume in HLA federations include the federation user/sponsor; the federation manager; technologists; security analysts; verification, validation, and accreditation (VV&A) agents; functional area experts; federation designers; execution planners; federation integrators; federation operators; federate representatives; and data analysts. Some roles (e.g., operators) are unique to a single activity in the process, while others are more pervasive throughout the process (e.g., federation manager). Since the applicability of a given role (as well as the set of activities it spans) varies from application to application, the activities described in this recommended practice specify the roles of individuals only in generic terms.

A major source of variation in how the seven-step process is implemented relates to the degree of reuse of existing federation products. In some cases, no previous work may exist, therefore a thoroughly original federation may need to be developed using a newly defined set of requirements to identify an appropriate set of federates and to build the full set of federation products needed to support an execution. In other cases, users of federations with established long-standing requirements will receive additional requirements. In this circumstance, the federation users can choose to reuse previous work, either in part or whole, along with the products of new developmental activities. In these situations, federation developers can often meet new user requirements by reusing a subset of an established core set of federates and defining appropriate modifications to other reusable federation products within their domain (e.g., FOM, planning documents). When an appropriate management structure exists to facilitate this type of federation development environment, significant savings can be achieved in both cost and development time.

The remainder of this recommended practice describes a structured, systems engineering approach to federation development and execution known as the HLA Federation Development and Execution Process (FEDEP). The seven-step process provides a top-level view of the FEDEP, while the FEDEP itself describes a decomposition of each of the seven major steps into a set of interrelated lower-level activities and supporting information resources. Since the needs of the HLA user community range from “first use” applications to experienced users, the FEDEP makes no assumptions about the existence of an established core set of federates or the up-front availability of reusable federation products. Although the intention is to define a comprehensive, generalized framework for HLA federation construction and execution, it is important to recognize that users of this process model will normally need to adjust and modify the FEDEP as appropriate to address the unique requirements and constraints of their particular application area. Users and developers of synthetic environments that are not based on HLA can also benefit from the guidance provided in this recommended practice, as the FEDEP can be augmented and/or modified as necessary to support nearly any type of distributed simulation application.

5. FEDEP model: detailed view

The FEDEP Model describes a high-level framework for the development and execution of HLA federations. The intent of the FEDEP Model is to specify a set of guidelines for federation development and execution that federation stakeholders can leverage to achieve the needs of their application.

A detailed view of the FEDEP Model is provided in Figure 2. This view illustrates the flow of information across the seven process steps identified in Figure 1. Data flow diagram notation is used in Figure 2 and throughout this recommended practice to represent activities (rounded rectangles), data stores (cylinders), and information flows and products (arrows) (see Scrudder, et al. [B4]).

The subclauses in Clause 5 describe the lower-level activities associated with each of the seven major federation development and execution steps. A tabular view of the activities inherent to each major step is provided in Table 1. Each activity description includes potential inputs and outputs of that activity and a representative list of recommended tasks. Graphical illustrations of the interrelationships among the activities within each step are also provided. Whenever outputs from one FEDEP activity represent a major input to one or more other activities, the arrow labels explicitly identify the activities that use these outputs. The arrow labels also identify the activities that produce inputs. However, there is a presumption embodied within the FEDEP that once a product has been created, it will be available for all subsequent activities, even though the product may not be shown as a major input or identified as an input in the activity description. Additionally, once a product is developed, the product may be modified or updated by subsequent activities without such modifications being explicitly identified either as a task or output. Input and output arrows without activity number labels are those in which the information originates from outside or is used outside the scope of the FEDEP.

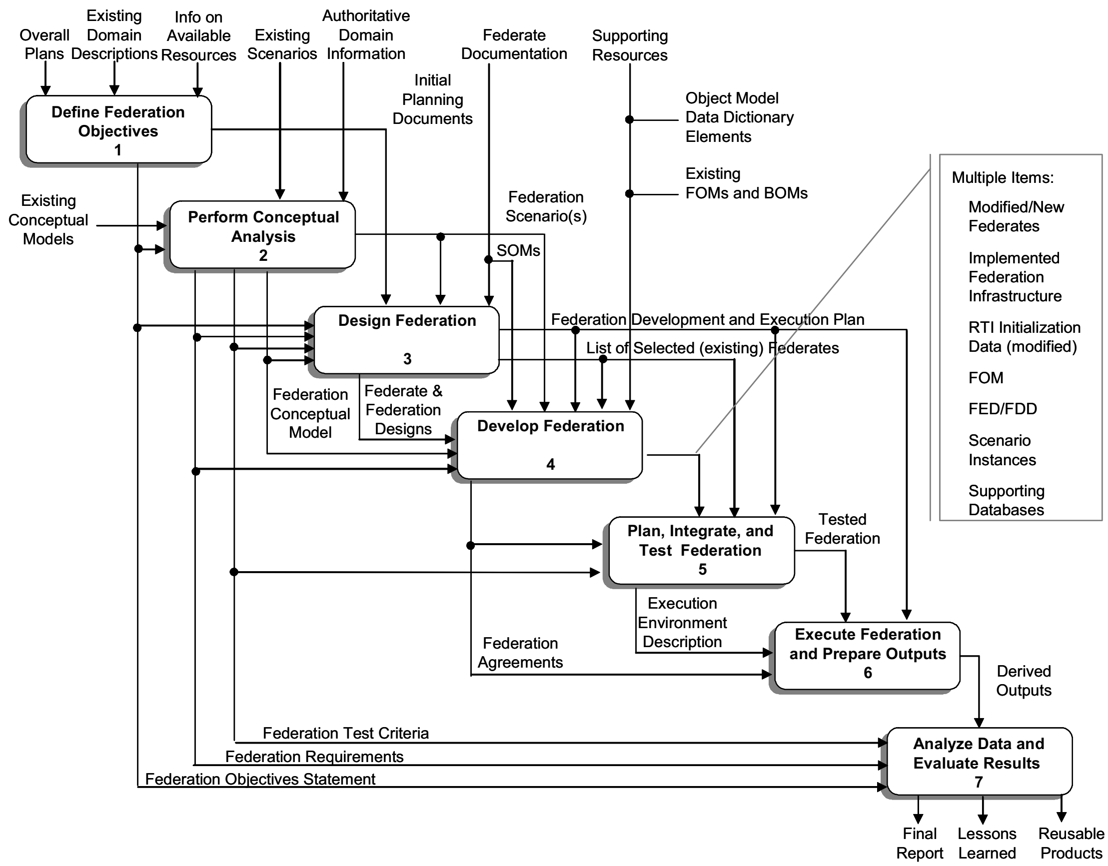

Figure 2—FEDEP, detailed view

Although many of the activities represented in the FEDEP diagram appear highly sequential, the intention is not to suggest a strict waterfall approach to federation development and execution. Rather, this process illustration is simply intended to highlight the major activities that occur during federation development and execution and approximately when such activities are first initiated relative to other federation development activities. In fact, experience has shown that many of the activities shown in Figure 2 as sequential are actually cyclic and/or concurrent, as was indicated earlier in Figure 1 via the dotted feedback arrows. Users of the FEDEP should be aware that the activities described in this recommended practice, while being generally applicable to most HLA federations, are intended to be tailored to meet the needs of each individual application. For example, FEDEP users should not feel constrained by the federation products explicitly identified in this recommended practice, but rather should produce whatever additional documentation is necessary to support their application. The guidance provided in this recommended practice should be used as a starting point for developing the specific approach to federation development and execution for the intended application.

Table 1—Tabular view of the FEDEP

Step 1: Define federation objectives	Step 2: Perform conceptual analysis	Step 3: Design federation	Step 4: Develop federation	Step 5: Plan, integrate, and test federation	Step 6: Execute federation and prepare outputs	Step 7: Analyze data and evaluate results
Identify user/ sponsor needs Develop objectives	Develop scenario Develop federation conceptual model Develop federation requirements	Select federates Prepare federation design Prepare plan	Develop FOM Establish federation agreements Implement federate designs Implement federation infrastructure	Plan execution Integrate federation Test federation	Execute federation Prepare federation outputs	Analyze data Evaluate and feedback results

5.1 Step 1: Define federation objectives

The purpose of Step 1 of the FEDEP is to define and document a set of needs that are to be addressed through the development and execution of an HLA federation and to transform these needs into a more detailed list of specific federation objectives.

Figure 3 illustrates the key activities in this step of the FEDEP. In this diagram (and all subsequent diagrams in this clause), each individual activity is labeled by a number designation (X.Y) to show traceability between the activity (Y) and the step (X) in the seven-step process to which the activity is associated. The activity number in these diagrams is intended only as an identifier and does not prescribe a particular ordering. Subclauses 5.1.1 through 5.1.2 describe each of these activities.

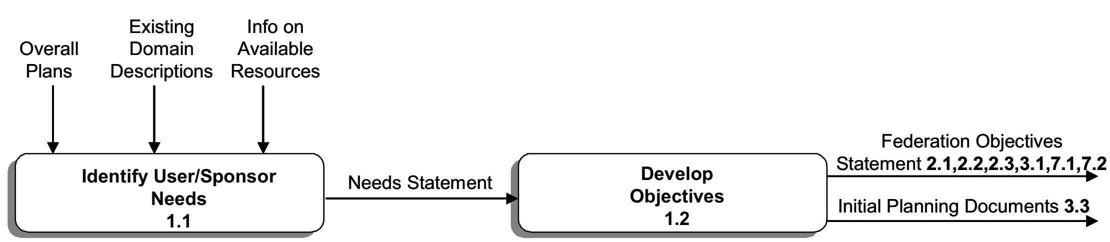

Figure 3—Define federation objectives (Step 1)

5.1.1 Activity 1.1: Identify user/sponsor needs

The primary purpose of this activity is to develop a clear understanding of the problem to be addressed by the federation. The needs statement may vary widely in terms of scope and degree of formalization. It should include, at a minimum, high-level descriptions of critical systems of interest, initial estimates of required fidelity and required behaviors for simulated entities, key events that must be represented in the federation scenario, and output data requirements. In addition, the needs statement should indicate the resources that will be available to support the federation (e.g., funding, personnel, tools, facilities) and any known constraints that may affect how the federation is developed (e.g., required federation participants, due dates, site and federation management requirements, and security requirements). In general, the needs statement should include as much detail and specific information as is possible at this early stage of the FEDEP.

An explicit and unambiguous statement of federation needs is critical to achieving clear communication of intent among the developers of the federation. Failure to establish a common understanding of the required product can result in costly rework in later stages of the FEDEP.

5.1.1.1 Activity inputs

The potential inputs to this activity are listed below. Neither this list of inputs, nor any subsequent lists of inputs, is meant to be completely exhaustive, nor are all mandatory for all federations.

- Overall plans (from the stakeholder's perspective).
- Existing domain descriptions.
- Information on available resources.

5.1.1.2 Recommended tasks

The potential tasks for this activity are listed below. Neither this list of tasks, nor any subsequent lists of tasks, is meant to be completely exhaustive, nor are all mandatory for all federations.

- Analyze the program objectives to identify the specific purpose and objective(s) that motivate development and execution of a federation.
- Identify available resources and known development and execution constraints.
- Document the information listed above in a needs statement.

5.1.1.3 Activity outcomes

The potential outcomes for this activity are listed below. Neither this list of outcomes, nor any subsequent lists of outcomes, is meant to be completely exhaustive, nor are all mandatory for all federations.

- Needs statement, including:
  - Federation purpose.
  - Identified needs (e.g., domain area/issue descriptions, high-level descriptions of critical systems of interest, initial estimates of required fidelity, and required behaviors for simulated players).
  - Key events that must be represented in a federation scenario.
  - Output data requirements.
  - Resources that will be available to support the federation (e.g., funding, personnel, tools, facilities).
  - Any known constraints which may affect how the federation is developed and executed (e.g., due dates, security requirements).

5.1.2 Activity 1.2: Develop objectives

The purpose of this activity is to refine the needs statement into a more detailed set of specific objectives for the federation. The federation objectives statement is intended as a foundation for generating federation requirements, i.e., translating high-level user/sponsor expectations into more concrete, measurable federation goals. This activity requires close collaboration between the federation user/sponsor and the federation development team to ensure that the original needs statement is properly analyzed and interpreted correctly, and that the resulting objectives are consistent with the stated needs.

Early assessments of federation feasibility and risk should also be performed as part of this activity. In particular, certain objectives may not be achievable given practical constraints (such as cost, schedule, availability of personnel or facilities) or even limitations on the state-of-the-art of needed technology. Early identification of such issues and consideration of these limitations and constraints in the federation objectives statement will set appropriate expectations for the federation development and execution effort.

Finally, the issue of tool selection to support scenario development, conceptual analysis, verification and validation (V&V), test activities, and configuration management should be addressed before the "develop objectives" activity is concluded. These decisions are made by the federation development team on the basis of tool availability, cost, applicability to the given application, and the personal preferences of the participants. The ability of a given set of tools to exchange federation data is also an important consideration.

5.1.2.1 Activity inputs

Potential inputs to this activity include the following:

— Needs statement.

5.1.2.2 Recommended tasks

Potential tasks for this activity include the following:

— Analyze the needs statement.
— Assess federation feasibility and risk.
— Define and document a prioritized set of federation objectives, consistent with the needs statement.
— Meet with the federation sponsor to review the federation objectives, and reconcile any differences.
— Define and document an initial federation development and execution plan.
— Identify potential tools to support the initial plan.

5.1.2.3 Activity outcomes

Potential outcomes for this activity include the following:

— Federation objectives statement, including:
  — Potential solution approaches and rationale for the selection of an HLA federation as the best approach.
  — A prioritized list of measurable objectives for the federation.
  — A high-level description of key federation characteristics (repeatability, portability, time management approach, availability, etc.).
  — Domain context constraints or preferences, including object actions/relationships, geographical regions, and environmental conditions.
  — Identification of federation execution constraints to include functional (e.g., federation execution control, federate execution control), technical (e.g., site, computational and network operations, federation health/performance monitoring), economic (e.g., available funding), and political (e.g., organizational responsibilities).
  — Identification of security needs, including probable security level and possible designated approval authority (or authorities, if a single individual is not possible).
  — Identification of key evaluation measures to be applied to the federation.

— Initial planning documents, including:
  — Federation development and execution plan showing an approximate schedule and major milestones.
  — Estimates of needed equipment, facilities, and data.
  — Initial planning documents for VV&A, test, configuration management, and security.
  — Tool selection to support scenario development, conceptual analysis, V&V, test activities, and configuration management.

5.2 Step 2: Perform conceptual analysis

The purpose of this step of the FEDEP is to develop an appropriate representation of the real world domain that applies to the federation problem space and to develop the federation scenario. It is also in this step that federation objectives are transformed into a set of highly-specific federation requirements that will be used in federation design, development, testing, execution, and evaluation. Figure 4 illustrates the key activities in this step of the FEDEP. Subclauses 5.2.1 through 5.2.3 describe each of these activities in detail.

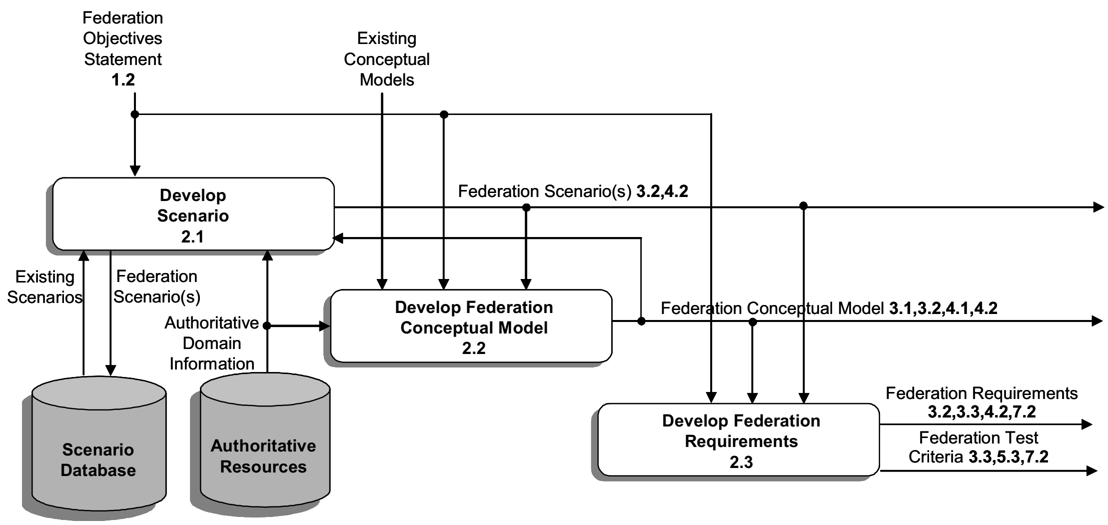

Figure 4—Perform conceptual analysis (Step 2)

5.2.1 Activity 2.1: Develop scenario

The purpose of this activity is to develop a functional specification of the federation scenario. Depending on the needs of the federation, the federation scenario may actually include multiple scenarios, each consisting of one or more temporally ordered sets of events and behaviors (i.e., vignettes). The primary input to this activity is the domain constraints specified in the federation objectives statement (Step 1), although existing scenario databases may also provide a reusable starting point for scenario development. Where appropriate, authoritative sources for descriptions of major entities and their capabilities, behavior, and relationships should be identified prior to scenario construction. A federation scenario includes the types and numbers of major entities that must be represented by the federation, a functional description of the capabilities, behavior, and relationships between these major entities over time, and a specification of relevant environmental conditions that impact or are impacted by entities in the federation. Initial conditions (e.g., geographical positions for physical objects), termination conditions, and specific geographic regions should also be provided. The product of this activity is a federation scenario or scenarios, which provides a bounding mechanism for conceptual modeling activities.

The presentation style used during scenario construction is at the discretion of the federation developers. Textual scenario descriptions, event-trace diagrams, and graphical illustrations of geographical positions for physical objects and communication paths all represent effective means of conveying scenario information. Software tools that support scenario development can generally be configured to produce these presentation forms. Reuse of existing scenario databases may also facilitate the scenario development activity.

5.2.1.1 Activity inputs

Potential inputs to this activity include the following:

- Federation objectives statement.
- Existing scenarios.
- Federation conceptual model.
- Authoritative domain information.

5.2.1.2 Recommended tasks

Potential tasks for this activity include the following:

- Choose the appropriate tool(s)/technique(s) for development and documentation of the federation scenario(s).
- Identify, using authoritative domain information, the entities, behaviors, and events that need to be represented in the federation scenario(s).
- Define one or more representative vignettes of federation events that, once executed, will produce the data necessary to achieve federation objectives.
- Define geographic areas of interest.
- Define environmental conditions of interest.
- Define initial conditions and termination conditions for the federation scenario(s).
- Ensure that an appropriate scenario (or scenario set) has been selected, or if new scenario information is to be developed, ensure with the stakeholder that the new scenario(s) will be acceptable.

5.2.1.3 Activity outcomes

Potential outcomes for this activity include the following:

- Federation scenario(s), including:
  - Types and numbers of major entities/objects that must be represented by the federation.
  - Description of entity/object capabilities, behaviors, and relationships.
  - Event timelines.
  - Geographical region(s).
  - Natural environment condition(s).
  - Initial conditions.
  - Termination conditions.

5.2.2 Activity 2.2: Develop federation conceptual model

During this activity, the federation development team produces a conceptual representation of the intended problem space based on their interpretation of user needs and federation objectives. The product resulting from this activity is known as a federation conceptual model (see Figure 4). The federation conceptual model provides an implementation-independent representation that serves as a vehicle for transforming federation objectives into functional and behavioral descriptions for system and software designers. The model also provides a crucial traceability link between the stated federation objectives and the eventual design implementation. This model can be used as the structural basis for many federation design and development activities (including scenario development) and can highlight correctable problems early in the federation development process when validated by the user/sponsor.

The federation conceptual model starts as a description of the entities and actions that need to be included in the federation in order to achieve all federation objectives. At this point, these entities and actions are described without any reference to the specific simulations that will be used in the federation. The federation conceptual model also contains an explanatory listing of the assumptions and limitations, which bound the model. In later steps of the FEDEP, the federation conceptual model transitions through additional enhancement into a reference product suitable for federation design.

The early focus of federation conceptual model development is to identify federation entities, to identify static and dynamic relationships between entities, and to identify the behavioral and transformational (algorithmic) aspects of each entity. Static relationships can be expressed as ordinary associations, or as more specific types of associations such as generalizations ("is-a" relationships) or aggregations ("part-whole" relationships). Dynamic relationships should include (if appropriate) the specification of temporally ordered sequences of entity interactions with associated trigger conditions. Entity characteristics (attributes) and interaction descriptors (parameters) may also be identified to the extent possible at this early stage of the process. While a conceptual model may be documented using differing notations, it is important that the conceptual model provides insight into the real world domain.

The federation conceptual model needs to be carefully evaluated before the next step (design federation) is begun, including a review of key processes and events by the user/sponsor to ensure the adequacy of the domain representation. Revisions to the original federation objectives and federation conceptual model may be defined and implemented as a result of this feedback. As the federation conceptual model evolves, it is transformed from a general representation of the real world domain to a more specific articulation of the capabilities of the federation as constrained by the federates and available resources. The federation conceptual model will serve as a basis for many later development activities such as federate selection, federation design, implementation, test, evaluation, and validation.

5.2.2.1 Activity inputs

Potential inputs to this activity include the following:

- Federation objectives statement.
- Authoritative domain information.
- Federation scenario(s).
- Existing conceptual models.

5.2.2.2 Recommended tasks

Potential tasks for this activity include the following:

- Choose the technique and format for development and documentation of the federation conceptual model.
- Identify and describe all relevant entities within the domain of interest.
- Define static and dynamic relationships between federation entities.
- Identify events of interest within the domain, including temporal relationships.
- Document the federation conceptual model and related decisions.
- Working with federation stakeholders, verify the contents of the conceptual model.

5.2.2.3 Activity outcomes

Potential outcomes for this activity include the following:

- Federation conceptual model.

5.2.3 Activity 2.3: Develop federation requirements

As the federation conceptual model is developed, it will lead to the definition of a set of detailed federation requirements. These requirements, based on the original federation objectives statement (Step 1), should be testable and should provide the implementation level guidance needed to design and develop the federation. The federation requirements should consider the specific execution management needs of all federation users, such as federation execution control, federate and federation monitoring, federation data logging, etc. Such needs may also impact the scenario developed in Activity 2.1. The federation requirements should also explicitly address the issue of fidelity, so that fidelity requirements can be considered during selection of federation participants. In addition, any programmatic or technical constraints on the federation should be refined and described to the degree of detail necessary to guide federation implementation.

5.2.3.1 Activity inputs

Potential inputs to this activity include the following:

- Federation objectives statement.
- Federation scenario(s).
- Federation conceptual model.

5.2.3.2 Recommended tasks

Potential tasks for this activity include the following:

- Define required behaviors of federation entities and required characteristics of federation events.
- Define requirements for live, virtual, and constructive simulations.
- Define human or hardware in-the-loop requirements.
- Define federation performance requirements.
- Define federation evaluation requirements.
- Define time management requirements (real-time versus slower or faster than real-time).
- Define host computer and networking hardware requirements.
- Define supporting software requirements.
- Define security requirements for hardware, network, data, and software.
- Define federation output requirements, including requirements for data collection and data analysis.
- Define execution management requirements.
- Ensure that federation requirements are clear, unique, and testable.
- Demonstrate traceability between federation requirements and program objectives, federation objectives, federation scenario(s), and federation conceptual model.
- Document all federation requirements.

5.2.3.3 Activity outcomes

Potential outcomes for this activity include the following:

- Federation requirements.
- Federation test criteria.

5.3 Step 3: Design federation

The purpose of this step of the FEDEP is to produce the design of the federation that will be implemented in Step 4. This involves identifying existing federation participants (federates) that are suitable for reuse, creating new federates and federate components if required, allocating the required functionality to federates, and developing a detailed plan for federation development and implementation. Figure 5 illustrates the key activities in this step of the FEDEP. Subclauses 5.3.1 through 5.3.3 describe each of these activities in detail.

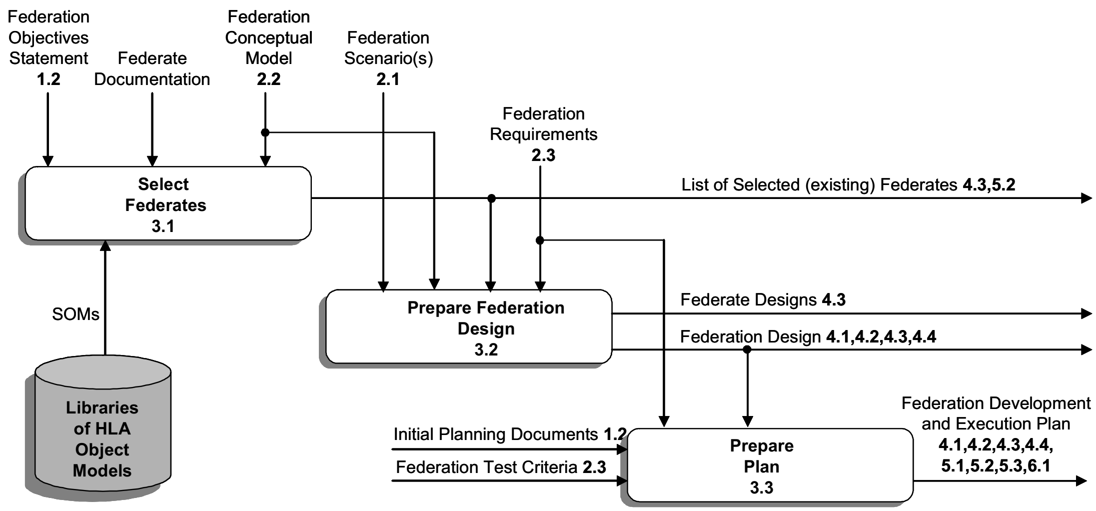

Figure 5—Design federation (Step 3)

5.3.1 Activity 3.1: Select federates

The purpose of this activity is to determine the suitability of individual simulation systems to become members of the federation. This is normally driven by the perceived ability of potential federation members to represent objects, activities, and interactions in the federation conceptual model. In some cases, these potential federation members may be federations themselves, such as an aircraft federate built from separate simulations of its subsystems. Managerial constraints (e.g., availability, security, facilities) and technical constraints (e.g., VV&A status, portability) may influence the selection of federation members.

In some federations, the identity of at least some federation participants will be known very early in the process. For instance, the federation sponsor may explicitly require the use of certain federates in the federation, or an existing federation (with well-established federates) may be reused and extended as necessary to address a new set of requirements. Although early federate selection may have certain advantages, it also introduces some immediate constraints on what the federation will and will not be able to do. Since required federate-level capabilities are not always well understood at the initiation of the federation development, it is generally advisable to defer final decisions on federation membership until this point in the overall process.

Libraries of HLA OMs may be searched for candidate federates, keyed to critical entities and actions of interest. To support final federate selection decisions, additional information resources (such as design and compliance documents) are generally necessary to fully understand internal simulation representations of required behaviors/activities and other practical aspects of federate utilization.

5.3.1.1 Activity inputs

The potential inputs to this activity include the following:

- Federation objectives statement.
- Federation conceptual model.
- Federate documentation [including Simulation Object Models (SOMs)].

5.3.1.2 Recommended tasks

The potential tasks for this activity include the following:

- Define criteria for federate selection.
- Determine if an existing, reusable federation meets or partially satisfies the federation requirements.
- Identify candidate federates, including predefined federation participants.
- Analyze the ability of each candidate federate to represent required federation entities/objects and events.
- Review federation purpose and objectives with respect to selected federates and availability of resources.
- Document rationale (including assumptions) for selection of federates.

5.3.1.3 Activity outcomes

The potential outcomes for this activity include the following:

- List of selected (existing) federates, including documented federate selection rationale.

5.3.2 Activity 3.2: Prepare federation design

Once all federates have been identified, the next major activity is to prepare the federation design and allocate the responsibility to represent the entities and actions in the federation conceptual model to the federates. This activity will allow for an assessment of whether the set of selected federates provides the full set of required functionality. A by-product of the allocation of functionality to the federates will be additional design information which can embellish the federation conceptual model.

As agreements on assigned responsibilities are negotiated, various federation design trade-off investigations may be conducted as appropriate to support the development of the federation design. Many of these investigations can be considered to be early execution planning and may include technical issues such as time management, federation management, infrastructure design, runtime performance, and potential implementation approaches. The major inputs to this activity include the federation requirements, the federation scenario, and the federation conceptual model (see Figure 5). In this activity, the federation conceptual model is used as a conduit to ensure that user domain-specific requirements are appropriately translated into the federation design. High-level federation design strategies, including modeling approaches and/or tool selection, may be revisited and renegotiated at this time based on inputs from the federates. When the federation represents a modification or extension to a previous federation, new federates must be made cognizant of all previously negotiated agreements within that earlier federation and given the opportunity to revisit pertinent technical issues. For secure federations, efforts associated with maintaining a secure posture during the federation execution can begin, including the designation of security responsibility. The initial security risk assessment and concept of operations may be refined at this time to clarify the security level and mode of operation.

In the case that an existing set of federates cannot fully address all federation requirements, it may be necessary to perform an appropriate set of design activities at the federate level. This may involve enhancements to one or more of the selected federates, or could even involve designing an entirely new federate. The federation development team must balance long-term reuse potential versus time and resource constraints in the evaluation of viable design options.

5.3.2.1 Activity inputs

The potential inputs to this activity include the following:

- Federation conceptual model.
- Federation scenario(s).
- Federation requirements.
- List of selected (existing) federates.

5.3.2.2 Recommended tasks

The potential tasks for this activity include the following:

- Analyze selected federates and identify those federates that best provide required functionality and fidelity.
- Allocate functionality to selected federates and determine if federate modifications are necessary and/or if development of a new federate(s) is needed.
- Develop design for needed federate modifications.
- Develop design for new federates (as necessary).
- Ensure that earlier federation decisions do not conflict with selected federates.
- Evaluate alternative federation design options, and identify the design that best addresses federation requirements.
- Develop design for federation infrastructure.
- Develop design of supporting databases.
- Estimate federation performance, and determine if actions are necessary to meet performance requirements.
- Analyze, and if necessary, refine initial security risk assessment and concept of operations.
- Document the federation design.

5.3.2.3 Activity outcomes

The potential outcomes for this activity include the following:

- Federation design, including:
  - Federate responsibilities.
  - Federation architecture (including supporting infrastructure design).
  - Supporting tools (e.g., RTI, performance measurement equipment, network monitors).
  - Implied requirements for federate modifications and/or development of new federates.

- Federate designs.

5.3.3 Activity 3.3: Prepare plan

Another major activity in Step 3 (federation design) is to develop a coordinated plan to guide the development, test, and execution of the federation. This requires close collaboration among all federation participants to ensure a common understanding of federation goals and requirements and also to identify (and agree to) appropriate methodologies and procedures based on recognized systems engineering principles. The initial planning documents prepared during development of the federation objectives provide the basis for this activity (see Figure 5). The plan should include the specific tasks and milestones for each federate, along with proposed dates for completion of each task.

The plan may also identify the software tools that will be used to support the remaining life cycle of the federation [e.g., RTI selection, federation runtime tools, Computer-Aided Software Engineering (CASE), configuration management, V&V, testing]. For federations with stochastic properties, (e.g., Monte Carlo techniques), the plan may include an experimental design. For new federations, a plan to design and develop a network configuration may be required. These agreements, along with a detailed work plan, must be documented for later reference and possible reuse in other federations.

5.3.3.1 Activity inputs

The potential inputs for this activity include the following:

- Initial planning documents.
- Federation requirements.
- Federation design.
- Federation test criteria.

5.3.3.2 Recommended tasks

The potential tasks for this activity include the following:

- Refine and augment the initial federation development and execution plan, including specific tasks and milestones for each federate.
- Identify needed federation agreements and plans for securing these agreements.
- Develop approach and plan for integration of the federation.
- Revise (as necessary) VV&A and test plans.
- Finalize plans for data collection, management, and analysis.
- Complete selection of supporting tools, and develop plan for acquiring and installing the tools.
- Develop plans and procedures for establishing and managing configuration baselines.
- Translate federation requirements into plans for federation execution and management.
- If required, prepare design of experiments.

5.3.3.3 Activity outcomes

The potential outcomes for this activity include the following:

- Federation development and execution plan, including:
  - Federation schedule, including detailed task and milestone identification.
  - Integration plan.
  - VV&A plan.
  - Test and evaluation plan.
  - Security plan.
  - Data management plan.
  - Configuration management plan.
  - Identification of required support tools.

5.4 Step 4: Develop federation

The purpose of this step of the FEDEP is to develop the FOM, modify federates if necessary, and prepare the federation for integration and test (database development, security procedure implementation, etc.). Figure 6 illustrates the key activities in this phase of the FEDEP. Subclauses 5.4.1 through 5.4.4 describe each of these activities in detail.

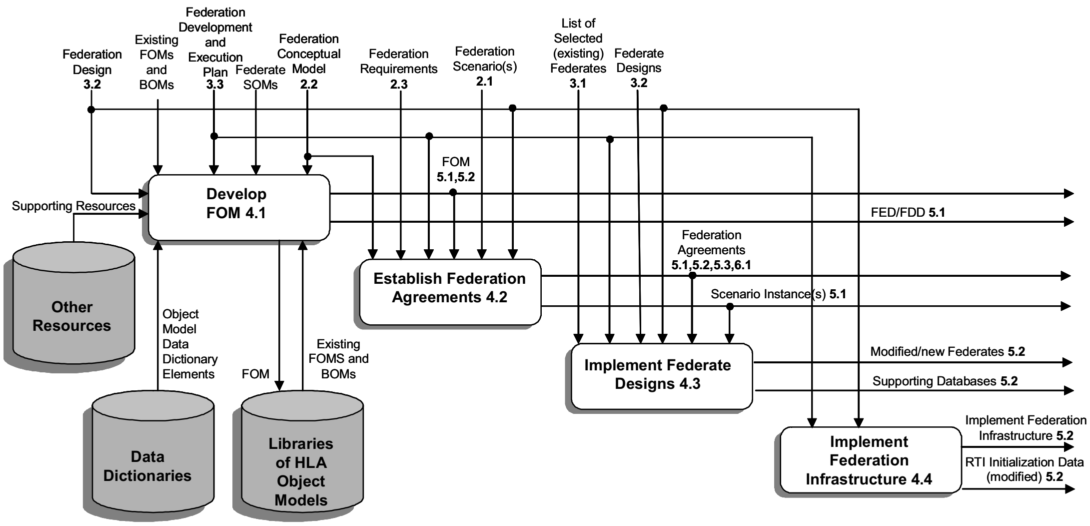

Figure 6—Develop federation (Step 4)

5.4.1 Activity 4.1: Develop FOM

Using federates identified to meet federation requirements, and the allocation of responsibilities for representation of entities and actions in the federation conceptual model across these federates, the FOM is developed to support the data exchanges required among the federates to meet the federation objectives. Several different fundamental approaches can be taken to FOM development, all of which have unique advantages depending on the particular situation. These approaches include the following:

— Construct the FOM using a “clean sheet” approach by using the federation scenario and federation conceptual model while applying any existing standards such as those found in a data dictionary.
— Merge together the SOMs of all participating federates, removing those aspects of the SOMs that do not apply to the domain of interest.
— Begin with the SOM that most closely aligns with the desired FOM, remove those aspects of the SOM that do not apply to the domain of interest, and merge in elements of other SOMs to fully represent the domain.
— Begin with a FOM(s) from a previous, but similar, application. Modify and/or augment as required.
— Begin with a FOM that provides a common frame of reference to a given user community. Remove elements of the FOM that are not required for the application. Modify and/or augment only if necessary.
— Employ a reusable set of OM components to construct and/or modify a FOM, with each component representing a single aspect of federation interplay. Base Object Models (BOMs) (see SISO-REF-005-2001 [B5]) are one example of such components.

While each of these last five approaches may represent a somewhat more efficient FOM development strategy (relative to starting entirely from scratch) under certain circumstances, all will require some use and appropriate tailoring of the essential activities described in the HLA OM Development Process (see [B1]). A summary of these activities is provided in Figure 7. Federation security personnel must always maintain knowledge of any classified information associated with applicable entries in each federate's SOM and the implications when this data is combined into a single FOM.

The use of libraries of HLA OMs and automated tools to facilitate the OM development process is strongly encouraged. Furthermore, OM libraries can provide users with access to reusable OMs that can be used as a template or common foundation for the development of a new FOM. These same libraries may also contain OM "piece parts" (e.g., individual classes, whole BOMs) that can be used as building blocks in the construction or augmentation of a FOM. Automated tools may be used to modify or extend an existing OM or to build an entirely new FOM. Such tools offer features such as consistency checking, syntax checking, Federation Execution Data (FED) generation (per HLA Version 1.3 specifications[5]) or FOM Document Data (FDD) generation (per IEEE Std 1516.1-2000), and online user manuals.

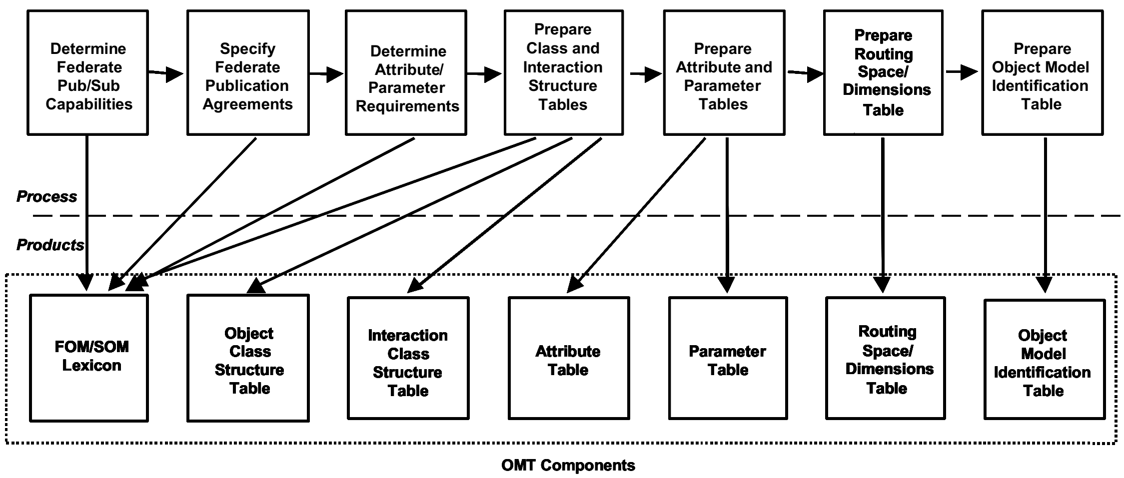

Figure 7—HLA OM development process

5.4.1.1 Activity inputs

The potential inputs to this activity include the following:

— Federation design.
— Federate SOMs.
— Federation development and execution plan.
— OM data dictionary elements.
— Existing FOMs and BOMs.
— Supporting resources (e.g., OM development tools, OM libraries, dictionaries).
— Federation conceptual model.

5.4.1.2 Recommended tasks

The potential tasks for this activity include the following:

— Choose a FOM development approach.
— Identify appropriate OMs or OM subsets for reuse.
— Review applicable data dictionaries to identify relevant OM elements.
— Develop and document the FOM using an appropriate tool.
— Verify that the FOM conforms to the federation conceptual model.

[5]Information on references can be found in Clause 2.

5.4.1.3 Activity outcomes

The potential outcomes for this activity include the following:

- FOM.
- FED/FDD.

5.4.2 Activity 4.2: Establish federation agreements

Although the FOM defines and documents the full set of data that is exchanged among federates to achieve federation objectives, there are other operating agreements that must be reached among federate developers and management (prior to implementation) that are not necessarily documented in the FOM. Such agreements are necessary to establish a fully consistent, interoperable, distributed simulation environment. While the actual process of establishing federation agreements begins early in the FEDEP and is embodied in each of its activities, this may not result in a complete set of formally documented agreements. It is at this point in the overall process that federation developers need to explicitly consider what additional agreements are required and how they should be documented.

There are many different types of federation agreements. For instance, federation members must use the federation conceptual model to gain an understanding and agreement on the behavior of all federation objects and how federation objects will interact with each other during the execution. While some of this information will be documented in the FOM, other means may be required to fully address these issues. Additional requirements for software modifications to selected federates may be identified as a result of these discussions; such requirements must be addressed prior to federation integration activities. Also, agreements must be reached as to the databases and algorithms that must be common (or at least consistent) across the federation to guarantee valid interactions among all federation participants. For instance, a consistent federation-wide view of simulated environment features and phenomena is critical in order for objects owned by different federates to interact and behave in a realistic fashion. In addition, certain operational issues must be addressed and resolved among the members of the federation. For instance, agreements on federation initialization procedures, synchronization points, save/restore policies, and security procedures are all necessary to ensure proper operation of the federation.

Once all authoritative data sources that will be used in support of the federation have been identified, the actual data stores are used to transition the functional description of the scenario (developed in Step 2; see Figure 4) to an executable scenario instance (or set of instances). The product of this activity permits federation testing to be conducted directly within the context of the domain of interest and also drives the execution of the federation later in the FEDEP.

Finally, federation developers must recognize that certain agreements may require the activation of other processes external to the FEDEP. For instance, utilization and/or modification of certain federates may require contractual actions between members of the federation or between the use/sponsor and the affected federate. Even where contractual actions may not be required, formal memoranda of agreement may be required between members. Additionally, federations requiring the processing of classified data will generally require the establishment of a security agreement between the federate security authorities. Each of these external processes has the potential to negatively impact the development and execution of a federation within resource and schedule constraints and should be factored into project plans as early as possible.

5.4.2.1 Activity inputs

The potential inputs to this activity include the following:

- Federation scenario(s).
- Federation conceptual model.
- Federation design.

— Federation development and execution plan.
— Federation requirements.
— FOM.

5.4.2.2 Recommended tasks

The potential tasks for this activity include the following:

— Decide the behavior of all federation objects and how they should interact during execution.
— Identify the necessary software modifications to selected federates, not previously identified.
— Decide which databases and algorithms must be common or consistent.
— Identify authoritative data sources for federate and federation databases.
— Build all required federate and federation databases.
— Decide how time should be managed in the federation.
— Establish synchronization points for the federation.
— Establish procedures for federation initiation.
— Decide strategy for how the federation should be saved and restored.
— Decide how data is to be distributed across the federation.
— Transform the functional scenario description to an executable scenario [scenario instance(s)].
— Review security agreements, and establish security procedures.

5.4.2.3 Activity outcomes

The potential outcomes for this activity include the following:

— Federation agreements, including:
  — Established security procedures.
  — Time management agreements.
  — Data management and distribution agreements.
  — Defined synchronization points.
  — Defined federation initialization procedures.
  — Federation save/restore strategy.
  — Agreements on supporting databases and algorithms.
  — Agreements on authoritative data sources.
  — Agreements on publication and subscription responsibilities.

— Scenario instance(s).

5.4.3 Activity 4.3: Implement federate designs

The purpose of this activity is to implement whatever modifications are necessary to the federates to ensure that they can represent assigned objects and associated behaviors as described in the federation conceptual model (Step 2), produce and exchange federation data with other federates as defined by the FOM, and abide by the established federation agreements. This may require internal modifications to the federate to support assigned domain elements, or it may require modifications or extensions to the federate's HLA interface to support new FOM data structures or HLA services that were not supported in the past. In some cases (for non-HLA compliant federates) it may even be necessary to develop an HLA interface for the federate. In this situation, the federate must consider both the resource (e.g., time, cost) constraints of the immediate application as well as longer-term reuse issues in deciding the best overall strategy for completing the federate interface. In situations where entirely new federates are needed, the implementation of the federate design must take place at this time.

5.4.3.1 Activity inputs

The potential inputs to this activity include the following:

— Federation development and execution plan.
— List of selected (existing) federates.
— Federate designs.
— Federation design.
— Federation agreements.
— Scenario instance(s).

5.4.3.2 Recommended tasks

The potential tasks for this activity include the following:

— Implement federate modifications to support allocated functionality.
— Implement modifications of, or extensions to, the HLA interfaces of all federates.
— Develop the HLA interface for non-HLA-compliant federates.
— Implement design of new federates as required.
— Implement design of supporting databases and scenario instance(s).
— Complete HLA compliance certification process (if required).

5.4.3.3 Activity outcomes

The potential outcomes for this activity include the following:

— Modified and/or new federates.
— Supporting databases.

5.4.4 Implement federation infrastructure

The purpose of this activity is to implement, configure, and initialize the infrastructure necessary to support the federation and verify that it can support the execution and intercommunication of all federation components. This involves the implementation of the network design, e.g., Wide Area Networks (WANs), Local Area Networks (LANs); the initialization and configuration of the network elements, e.g., routers, bridges; and the installation and configuration of supporting software on all computer systems. This also involves whatever facility preparation is necessary to support integration and test activities.

In situations in which federation performance is an especially critical issue, it may be desirable to modify the RTI initialization data associated with the specific RTI implementation being used in the federation. Although extensive modifications to the RTI initialization data are generally unnecessary, and should only be undertaken with sufficient knowledge of their potential impacts on the federation as a whole, minor modifications can improve federation performance in some circumstances.

5.4.4.1 Activity inputs

The potential inputs to this activity include the following:

— Federation design.
— Federation development and execution plan.

5.4.4.2 Recommended tasks

The potential tasks for this activity include the following:

- Prepare integration/test facility, including:
  - Ensure basic facility services (air conditioning, electric power, etc.) are functional and available.
  - Ensure availability of required hardware/software in integration/test facility.
  - Perform required system administration functions (establish user accounts, establish procedures for file backups, etc.).

- Implement infrastructure design, including:
  - Install and configure required hardware elements.
  - Install and configure RTI and other supporting software.
  - Test infrastructure to ensure proper operation.

5.4.4.3 Activity outcomes

The potential outcomes for this activity include the following:

- Implemented federation infrastructure.
- Modified (if necessary) RTI initialization data.

5.5 Step 5: Plan, integrate, and test federation

The purpose of this step of the FEDEP is to plan the federation execution, establish all required interconnectivity between federates, and test the federation prior to execution. Figure 8 illustrates the key activities in this step of the FEDEP. Subclauses 5.5.1 through 5.5.3 describe each of these activities.

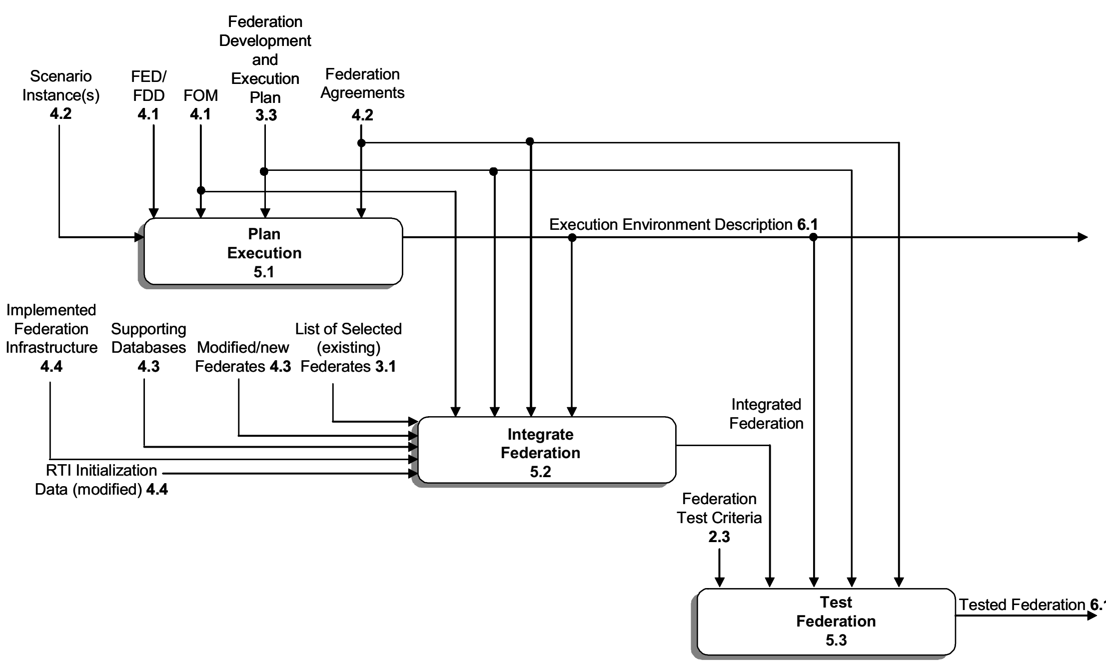

Figure 8—Plan, integrate, and test federation (Step 5)

5.5.1 Activity 5.1: Plan execution

The main purpose of this activity is to fully describe the federation execution environment and develop an execution plan. For instance, federate/federation performance requirements and salient characteristics of host computers, operating systems, and networks that will be used in the federation should all be documented at this time. The completed set of information, taken together with the FOM and associated FED/FDD, provides the necessary foundation to transition into the integration and testing phase of federation development.

Additional activities in this step include the incorporation of any necessary refinements to federation test and VV&A plans, and (for secure federations) the development of a security test and evaluation plan. This latter activity requires reviewing and verifying the security work accomplished thus far in the federation development and finalizing the technical details of security design, such as information downgrading rules, formalized practices, etc. This plan represents an important element of the necessary documentation set for the federation.

Operational planning is also a key aspect of this activity. This planning should address who will be involved in each execution run, both in a support and operational role. It should detail the schedule for both the execution runs and the necessary preparation prior to each run. Training and rehearsal for federation support and operational personnel should be addressed as necessary. Specific procedures for starting, conducting, and terminating each execution run should be documented.

5.5.1.1 Activity inputs

The potential inputs to this activity include the following:

— FOM.
— FED/FDD.
— Scenario instance(s).
— Federation agreements.
— Federation development and execution plan.

5.5.1.2 Recommended tasks

The potential tasks for this activity include the following:

— Refine/augment federation development and execution plan in the areas of VV&A, test, and security, as necessary.
— Assign federation components to appropriate infrastructure elements.
— Identify risks, and take action to reduce risks.
— Document all information relevant to the federation execution.
— Develop detailed execution plans.

5.5.1.3 Activity outcomes

The potential outcomes for this activity include the following:

— Execution environment description.

5.5.2 Activity 5.2: Integrate federation

The purpose of this activity is to bring all of the federation participants into a unifying operating environment. This requires that all federate hardware and software assets are properly installed and interconnected in a configuration that can satisfy all FOM data interchange requirements and federation agreements.

Because WAN/LAN problems are often difficult to diagnose and correct, the WAN/LAN connection should be established as a first step, especially when dealing with secure connections. The federation development plan specifies the methodology used in this activity for federation integration, and the federation scenario instance provides the necessary context for integration activities.

Federation integration is normally performed in close coordination with federation testing. Iterative "test-fix-test" approaches are used quite extensively in practical applications and have been shown to be quite effective.

5.5.2.1 Activity inputs

The potential inputs to this activity include the following:

— Federation development and execution plan.
— Execution environment description.
— Federation agreements.
— FOM.
— RTI initialization data.
— Federates (existing selected, modified and/or newly developed federates).
— Implemented federation infrastructure.
— Supporting databases.

5.5.2.2 Recommended tasks

The potential tasks for this activity include the following:

— Ensure that all federate software is properly installed and interconnected.
— Establish method for managing known software problems and "workarounds."
— Perform incremental federation integration according to plan.

5.5.2.3 Activity outcomes

The potential outcomes for this activity include the following:

— Integrated federation.

5.5.3 Activity 5.3: Test federation

The purpose of this activity is to test that all of the federation participants can interoperate to the degree required to achieve federation objectives. Three levels of testing are defined for HLA applications:

a) *Federate testing:* In this activity, each federate is tested to ensure that the federate software correctly implements the federation requirements as documented in the HLA FOM, execution environment description, and any other federation operating agreements.
b) *Integration testing:* In this activity, the federation is tested as an integrated whole to verify a basic level of interoperability. This testing primarily includes observing the ability of the federates to interact correctly with the RTI and to exchange data as described by the FOM.
c) *Federation testing:* In this activity, the ability of the federation to interoperate to the degree necessary to achieve federation objectives is tested. This includes observing the ability of federates to interact according to the defined scenario and to the level of fidelity required for the application. This activity also includes security certification testing if required for the application. The results from federation testing may contribute to verification and validation of the federation as required.

Procedures for conducting federation testing must be agreed upon by all federation participants and documented appropriately. Data collection plans should be exercised during the testing phase to ensure that the data needed to support the federation objectives is being accurately collected and stored. The HLA Management Object Model (MOM) may be used during integration/federation testing to provide useful information on the operation of the RTI, individual federates, and the integrated federation.

The desired output from this activity is an integrated, tested, validated, and if required, accredited federation that indicates that execution of the federation can commence. If early testing and validation uncover obstacles to successful federation integration and accreditation, federate or federation developers must take corrective actions. In many cases, these corrective actions simply require a relatively minor software fix (or series of fixes) or minor adjustment to the FOM. However, testing may also uncover more serious software, interoperability, or validity problems. In these cases, options may need to be identified, with their associated cost and schedule estimates (including security and VV&A implications), and should be discussed with the federation user/sponsor before corrective action is taken.

Finally, whenever a federate has modified its HLA interface to meet federation requirements, that federate should be tested (or retested) for compliance to the HLA. Although this task may be performed at this time, compliance testing may also be performed as a post-federation activity.

5.5.3.1 Activity inputs

The potential inputs to this activity include the following:

- Federation development and execution plan.
- Federation agreements.
- Execution environment description.
- Integrated federation.
- Federation test criteria.

5.5.3.2 Recommended tasks

The potential tasks for this activity include the following:

- Perform federate-level testing.
- Perform federation-level connectivity and interoperability testing.
- Analyze testing results (i.e., compare against test criteria).
- Review test results with federation user/sponsor.

5.5.3.3 Activity outcomes

The potential outcomes for this activity include the following:

- Tested (and if necessary, accredited) federation, including:
  - Federate test data.
  - Tested federates.
  - Federation test data.
  - Corrective actions.

5.6 Step 6: Execute federation and prepare outputs

The purpose of this step of the FEDEP is to execute the federation and to pre-process the output data from the federation execution. Figure 9 illustrates the key activities in this step of the FEDEP. Subclauses 5.6.1 through 5.6.2 describe each of these activities.

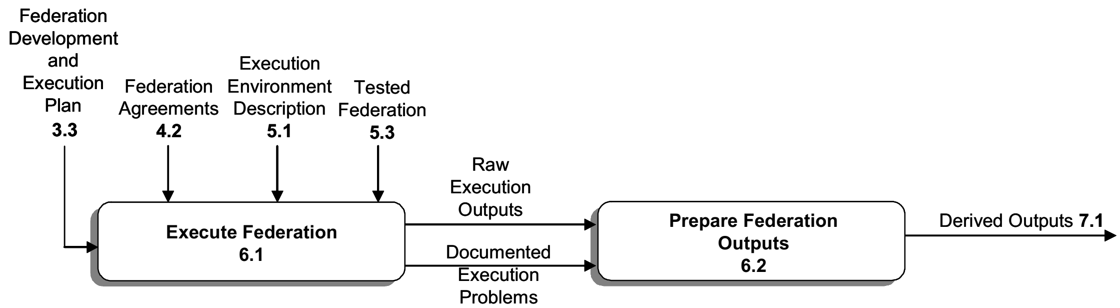

Figure 9—Execute federation and prepare outputs (Step 6)

5.6.1 Activity 6.1: Execute federation

The purpose of this activity is to exercise all federation participants in a coordinated fashion over time to generate required outputs, and thus achieve stated federation objectives. The federation must have been tested successfully before this activity can begin.

Execution management and data collection are critical to a successful federation execution. Execution management involves controlling and monitoring the execution via specialized software tools (as appropriate). Execution can be monitored at the hardware level [e.g., Central Processing Unit (CPU) usage, network load], and/or software operations can be monitored for individual federates or across the full federation. During execution, key federation test criteria should be monitored to provide an immediate evaluation of the successful execution of the federation.

Data collection is focused on assembling the desired set of outputs and on collecting whatever additional supporting data is required to assess the validity of the federation execution. In some cases, data is also collected to support replays of the federation execution (i.e., "playbacks"). Essential federation data may be collected via databases in the federates themselves, or can be collected via specialized data collection tools directly interfaced to the RTI. The particular strategy for data collection in any particular federation is entirely at the discretion of the federation development team, and should have been documented in the federation requirements, the federation development and execution plan, and in the federation agreements.

For secure federations, strict attention must be given to maintaining the security posture of the federation during execution. A clear concept of operations, properly trained security personnel, and strict configuration management will all facilitate this process. It is important to remember that authorization to operate (accreditation) is usually granted for a specific configuration of federates. Any change to the federates or federation composition will certainly require a security review and may require some or all of the security certification tests to be redone.

5.6.1.1 Activity inputs

The potential inputs to this activity include the following:

— Tested federation.
— Federation development and execution plan.
— Federation agreements.
— Execution environment description.

5.6.1.2 Recommended tasks

The potential tasks for this activity include the following:

- Perform identified executions and collect data.
- Manage the execution in accordance with the federation development and execution plan.
- Document detected problems during execution.
- Ensure continued secure operation in accordance with certification and accreditation decisions and requirements.

5.6.1.3 Activity outcomes

The potential outcomes for this activity include the following:

- Raw execution outputs (data).
- Documented execution problems.

5.6.2 Activity 6.2: Prepare federation outputs

The purpose of this activity is to pre-process the output collected during the federation execution prior to the formal analysis of the data in Step 7. This may involve the use of data reduction techniques to reduce the quantity of data to be analyzed and to transform the data to a particular format. Where data has been acquired from many sources, data fusion techniques may have to be employed. The data should be reviewed and appropriate action taken where missing or erroneous data is suspected. This may require further federation executions to be conducted.

5.6.2.1 Activity inputs

The potential inputs to this activity include the following:

- Raw execution outputs (data).
- Documented execution problems.

5.6.2.2 Recommended tasks

The potential tasks for this activity include the following:

- Merge data from multiple sources.
- Reduce/transform raw data.
- Review data for completeness and possible errors.

5.6.2.3 Activity outcomes

The potential outcomes for this activity include the following:

- Derived outputs.

5.7 Step 7: Analyze data and evaluate results

The purpose of this step of the FEDEP is to analyze and evaluate the data acquired during the federation execution (Step 6), and to report the results back to the user/sponsor. This evaluation is necessary to ensure that the federation fully satisfies the requirements of the user/sponsor. The results are fed back to the user/sponsor so that they can decide if the federation objectives have been met, or if further work is required. In the latter case, it will be necessary to repeat some of the FEDEP steps again with modifications to the appropriate federation products. Figure 10 illustrates the key activities in this step of the FEDEP. Subclauses 5.7.1 through 5.7.2 describe each of these activities.

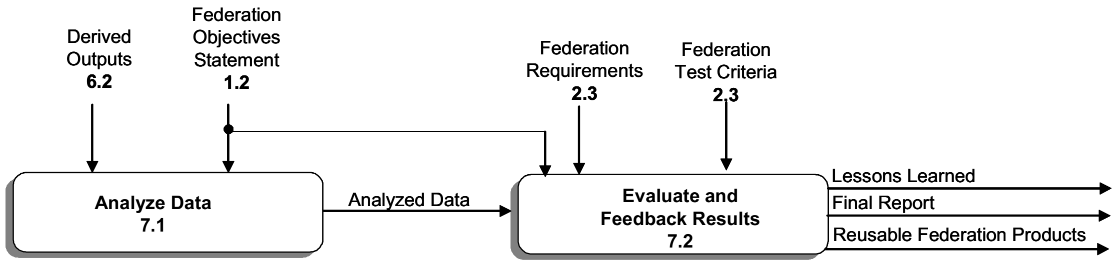

Figure 10—Analyze data and evaluate results (Step 7)

5.7.1 Activity 7.1: Analyze data

The main purpose of this activity is to analyze the derived outputs from Step 6. This data may be supplied using a range of different media (e.g., digital, video, audio), and appropriate tools and methods will be required for analyzing the data. These may be commercial off-the-shelf (COTS) or government off-the-shelf (GOTS) tools or specialized tools developed for a specific federation. The analysis methods used will be specific to a particular federation and can vary between simple observations (e.g., determining how many targets have been hit) to the use of complex algorithms (e.g., regression analysis or data mining). In addition to data analysis tasks, this activity also includes defining appropriate "pass/fail" evaluation criteria for the federation execution and defining appropriate formats for presenting results to the user/sponsor.

5.7.1.1 Activity inputs

The potential inputs to this activity include the following:

— Derived outputs.
— Federation objectives statement.

5.7.1.2 Recommended tasks

The potential tasks for this activity include the following:

— Apply analysis methods and tools to data.
— Define appropriate presentation formats.
— Prepare data in chosen formats.

5.7.1.3 Activity outcomes

The potential outcomes for this activity include the following:

— Analyzed data.

5.7.2 Activity 7.2: Evaluate and feedback results

The purpose of this activity is to determine if federation objectives have been met and to archive reusable federation products. There are two main tasks in this activity. In the first task, the derived results from the previous activity are evaluated to determine if all federation objectives have been met. This requires a retracing of execution results to the measurable set of federation requirements originally generated during conceptual analysis (Step 2) and refined in subsequent steps. This step also includes evaluating the results against the federation test criteria. In the vast majority of cases, any impediments to fully satisfying federation requirements have already been identified and resolved during the earlier federation development and integration phases. Thus, for well-designed federations, this task is merely a final check. In those rare cases in which certain federation objectives have not been fully met at this late stage of the overall process, corrective actions must be identified and implemented. This may necessitate revisiting previous steps of the FEDEP and regenerating federation results.

The second task in this activity, assuming all federation objectives have been achieved, is to store all reusable federation products in an appropriate archive for general reuse within the domain or broader HLA community. At a minimum, this would include archiving the FOM and any modifications to the SOMs of federation participants. However, there may be other federation products that may also be reusable, such as the federation scenario and the federation conceptual model. In fact, it may be advantageous in some instances to capture the full set of federation products required to reproduce the federation execution. Determination of which federation products have potential for reuse in future applications is at the discretion of the federation development team.

5.7.2.1 Activity inputs

The potential inputs to this activity include the following:

— Analyzed data.
— Federation objectives statement.
— Federation requirements.
— Federation test criteria.

5.7.2.2 Recommended tasks

The potential tasks for this activity include the following:

— Determine if all federation objectives have been met.
— Take appropriate corrective actions if deficiencies are found.
— Archive all reusable federation products.

5.7.2.3 Activity outcomes

The potential outcomes for this activity include the following:

— Lessons learned.
— Final report.
— Reusable federation products.

6. Conclusion

This recommended practice has provided a view of the FEDEP. Currently, this model represents the best practices available to the HLA community. The FEDEP is an easily tailored process and is offered as guidance to all participants in HLA activities. As additional experience is accrued in building HLA applications, the FEDEP will leverage this knowledge and evolve accordingly.

In the longer term, the FEDEP is expected to serve as a framework for the development of alternative, more detailed views of the FEDEP that may better satisfy the needs of specific communities. Such views can provide implementation level guidance to “hands-on” federation participants without the need to interpret and customize the more generalized FEDEP activity descriptions to a particular domain. Participants in HLA activities are encouraged to perform these types of adaptations whenever appropriate.

Annex A

(informative)

Bibliography

[B1] "HLA Object Model Development Process and Supporting Tools," *International Training and Education Conference (ITEC) Briefing*, Lausanne, Switzerland, Apr. 1998.[6]

[B2] IEEE 100™, *The Authoritative Dictionary of IEEE Standards Terms*, Seventh Edition.

[B3] IST-SP-96-01, A Glossary of Modeling and Simulation Terms for Distributed Interactive Simulation (DIS).[7]

[B4] Scrudder R., Waite W., Richardson M., and Lutz R., "Graphical Presentation of the Federation Development and Execution Process," 98F-SIW-103, *Simulation Interoperability Workshop*, Fall 1998.[8]

[B5] SISO-REF-005-2001, *Base Object Model (BOM) Study Group Final Report*, SISO Reference Product, Gustavson, P., et al, May 2001.[8]

[6]For more information, please visit http://www.itec.co.uk.

[7]For information on projects underway at the Institute for Simulation and Training at the University of Central Florida, please visit http://www.ist.ucf.edu.

[8]For more information, please visit http://www.sisostds.org.